Initialization

Code

```
Quit
```

```
Print[$MachineName ]
Print[$Version]
```

```
pcl0517
```

```
10.0 for Linux x86 (64-bit) (December  4, 2014)
```

```
<< RBSFA`
Print[$RBSFAversion]
Print[$RBSFAcommit  ]
```

```
RB-SFA v2.1.3, Tue 28 Feb 2017 11:48:14
```

```
commit   3bf93b827ba584ae1cc1b7f265f38a5f0a32ea04
Author: Emilio  Pisanty <emilio .pisanty@icfo.eu>
Date:   Tue Feb 28 11:49:15 2017 +0100
    Improved  testing for previous
     versions of ReIm  to avoid error messages  on package reload.
```

```
SystemOpen [$UserBaseDirectory]
```

```
$HistoryLength= (*1*) 0;
LaunchKernels[8 - $KernelCount];
ParallelEvaluate[{$MachineName , $RBSFAversion}] // Tally
```

```
{{{pcl0517, RB-SFA v2.1.3, Tue 28 Feb 2017 11:48:14}, 8}}
```

```
$HistoryLength= (*1*) 0;
LaunchKernels[4 - $KernelCount];
ParallelEvaluate[{$MachineName , $RBSFAversion}] // Tally
```

```
{{{cacomixtle , RB-SFA v2.1.3, Tue 28 Feb 2017 11:48:14}, 4}}
```

quick data import

```
<< (NotebookDirectory[] <> "data - saddle points on the rotating frame .txt");
<< (NotebookDirectory[] <> "data - SPA dipole data on the rotating frame .txt");
<< (NotebookDirectory[] <>
        "data - SPA dipole data for a single ionizationburst on the
            rotating frame .txt");
<< (NotebookDirectory[] <> "data - SPA dipole data for a single burst, with
            no intrinsicphase, on the rotating frame .txt");
saddlePoints // Dimensions
SPAdipoleData // Dimensions
singleSPAdipoleData // Dimensions
singleSPAdipoleDataNoIntrinsicPhase // Dimensions
```

```
{3, 1633}

{4, 1633}

{4, 1633}

{4, 1633}
```

Formatting niceties

```
<< MaTeX`
SetOptions[MaTeX, "Preamble " → {"
\\usepackage{amssymb ,upref}
\\usepackage{fourier}
\\usepackage{tgheros}
\\usepackage[T1]{fontenc}
\\usepackage{textcomp }
\\usepackage{microtype }
\\usepackage{siunitx}
\\usepackage{MnSymbol }
"}];
```

Formatting: label, tick and inset font sizes

```
lfs = 7;
tfs = 6;
ifs = 9;
ilfs = 5;
```

```
$OutputDirectory = FileNameJoin [{NotebookDirectory[], "..", "Figures"}];
$MainDirectory =
    StringReplace[FileNameJoin [{NotebookDirectory[], ".."}], {" " → "\\ "}];
pdflatex[] := Which[
    $OperatingSystem == "Unix", Run["cd " <> $MainDirectory <>
          " && pdflatex --output-directory=build Manuscript.tex"] /.
      {0 → "pdflatex successful"},
    $System == "Microsoft Windows (64-bit)", Run["cd " <> NotebookDirectory[] <>
          " && cd .. && pdflatex --output-directory=build Manuscript.tex"] /.
      {0 → "pdflatex successful"},
    True, Print["pdflatex[] has not been tested on Mac OS."];]
```

Definitions

Fields

```
bicircularA[t_] = F/ω {Cos[ω t], Sin[ω t], 0} + F/(2 ω) {Cos[2 ω t], -Sin[2 ω t], 0};
bicircularF[t_] = F {Sin[ω t], -Cos[ω t], 0} + F {Sin[2 ω t], Cos[2 ω t], 0};
bicircularArot[t_] = F/ω {3/2 Cos[3/2 ω t], 1/2 Sin[3/2 ω t], 0};
bicircularFrot[t_] = 2 F {Sin[3/2 ω t], 0, 0};
(* ({{Cos[α t], -Sin[α t], 0}, {Sin[α t], Cos[α t], 0}, {0, 0, 1}}) . (({Cos[ω t], Sin[ω t], 0}) + 1/2 ({Cos[2ω t], -Sin[2ω t], 0})) /. {α → ω/2} // FullSimplify
   ({{Cos[α t], -Sin[α t], 0}, {Sin[α t], Cos[α t], 0}, {0, 0, 1}}) . (({Sin[ω t], -Cos[ω t], 0}) + ({Sin[2ω t], Cos[2ω t], 0})) /. {α → ω/2} // FullSimplify *)
Rzα[t_] = {{Cos[ω/2 t], Sin[ω/2 t], 0}, {-Sin[ω/2 t], Cos[ω/2 t], 0}, {0, 0, 1}};
pi[{px_, py_, pz_}, t_, tt_] := {px, py, pz} + bicircularArot[t]
pi[{px_, py_, pz_}, t_] := pi[{px, py, pz}, t, 0]
```

Building the action

Saddle-point momentum

```
ps[t_, tt_] = -1/(t - tt) FullSimplify[Rzα[-tt].Integrate[bicircularA[τ], {τ, tt, t}]]
```

$$\left\{-\frac{1}{4\,(t-tt)\,\omega^2}F\left(-5\,\mathrm{Sin}\left[\frac{3\,tt\,\omega}{2}\right]+4\,\mathrm{Sin}\left[\frac{1}{2}(2\,t+tt)\,\omega\right]+\mathrm{Sin}\left[2\,t\,\omega-\frac{tt\,\omega}{2}\right]\right),\right.$$
$$\left.-\frac{1}{4\,(t-tt)\,\omega^2}F\left(3\,\mathrm{Cos}\left[\frac{3\,tt\,\omega}{2}\right]-4\,\mathrm{Cos}\left[\frac{1}{2}(2\,t+tt)\,\omega\right]+\mathrm{Cos}\left[2\,t\,\omega-\frac{tt\,\omega}{2}\right]\right),\ 0\right\}$$

Action

$$\texttt{S[t\_, tt\_]} = \left(\texttt{Ip} - \texttt{m}\ \frac{\omega}{2}\right)(\texttt{t}-\texttt{tt}) + ((\#/.\{\tau\to \texttt{t}\}) - (\#/.\{\tau\to \texttt{tt}\}))\ \&\texttt{@Integrate}\Big[\texttt{Total}\Big[\frac{1}{2}\left(\texttt{Rz}\alpha\texttt{[tt}-\tau\texttt{].ps[t, tt]} + \texttt{bicircularArot}[\tau]\right)^2\Big], \tau\Big]$$

$$(\texttt{t}-\texttt{tt})\left(\texttt{Ip}-\frac{\texttt{m}\ \omega}{2}\right)+ \frac{1}{32(\texttt{t}-\texttt{tt})^2\omega^4}\texttt{F}^2\Big(34\,\texttt{tt}+20\,\texttt{t}^3\omega^2-40\,\texttt{t}^2\,\texttt{tt}\,\omega^2+20\,\texttt{t}\,\texttt{tt}^2\omega^2+8\,\texttt{t}\,\texttt{Cos}[3\,\texttt{t}\,\omega]-16\,\texttt{tt}\,\texttt{Cos}[3\,\texttt{t}\,\omega]- 32\,\texttt{t}\,\texttt{Cos}[(\texttt{t}-\texttt{tt})\,\omega]-2\,\texttt{t}\,\texttt{Cos}[2\,(\texttt{t}-\texttt{tt})\,\omega]-8\,\texttt{t}\,\texttt{Cos}[3\,\texttt{tt}\,\omega]+32\,\texttt{t}\,\texttt{Cos}[(-\texttt{t}+\texttt{tt})\,\omega]- 32\,\texttt{tt}\,\texttt{Cos}[(-\texttt{t}+\texttt{tt})\,\omega]+2\,\texttt{t}\,\texttt{Cos}[2\,(-\texttt{t}+\texttt{tt})\,\omega]-2\,\texttt{tt}\,\texttt{Cos}[2\,(-\texttt{t}+\texttt{tt})\,\omega]+8\,\texttt{tt}\,\texttt{Cos}[(2\,\texttt{t}+\texttt{tt})\,\omega]+ 8\,\texttt{tt}\,\texttt{Cos}[(\texttt{t}+2\,\texttt{tt})\,\omega]+\frac{16}{3}\texttt{t}^2\,\omega\,\texttt{Sin}[3\,\texttt{t}\,\omega]-\frac{32}{3}\texttt{t}\,\texttt{tt}\,\omega\,\texttt{Sin}[3\,\texttt{t}\,\omega]+\frac{16}{3}\texttt{tt}^2\,\omega\,\texttt{Sin}[3\,\texttt{t}\,\omega]\Big)- \frac{1}{32(\texttt{t}-\texttt{tt})^2\omega^4}\texttt{F}^2\Big(34\,\texttt{t}+20\,\texttt{t}^2\,\texttt{tt}\,\omega^2-40\,\texttt{t}\,\texttt{tt}^2\omega^2+20\,\texttt{tt}^3\omega^2-8\,\texttt{tt}\,\texttt{Cos}[3\,\texttt{t}\,\omega]-32\,\texttt{t}\,\texttt{Cos}[(\texttt{t}-\texttt{tt})\,\omega]- 2\,\texttt{t}\,\texttt{Cos}[2\,(\texttt{t}-\texttt{tt})\,\omega]-16\,\texttt{t}\,\texttt{Cos}[3\,\texttt{tt}\,\omega]+8\,\texttt{tt}\,\texttt{Cos}[3\,\texttt{tt}\,\omega]+8\,\texttt{t}\,\texttt{Cos}[(2\,\texttt{t}+\texttt{tt})\,\omega]+ 8\,\texttt{t}\,\texttt{Cos}[(\texttt{t}+2\,\texttt{tt})\,\omega]+\frac{16}{3}\texttt{t}^2\,\omega\,\texttt{Sin}[3\,\texttt{tt}\,\omega]-\frac{32}{3}\texttt{t}\,\texttt{tt}\,\omega\,\texttt{Sin}[3\,\texttt{tt}\,\omega]+\frac{16}{3}\texttt{tt}^2\,\omega\,\texttt{Sin}[3\,\texttt{tt}\,\omega]\Big)$$

Building the prefactor

Ground-state quantities for a short-range potential

$$\texttt{shortRange}\Psi\texttt{[m\_ },\kappa\texttt{\_},\{\texttt{kx\_, ky\_, kz\_}\}]\ /;\ (\texttt{m}\ ^2 == 1) := \frac{2\,i}{\sqrt{2\pi}\,\kappa}\texttt{SolidHarmonicS}\,[1, \texttt{m}\ , \texttt{kx, ky, kz}]\,\frac{1}{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}\left(\frac{\kappa}{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2+\kappa^2}-\frac{\texttt{ArcTan}\left[\sqrt{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}\Big/\kappa\right]}{\sqrt{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}}\right)$$

$$\texttt{shortRange}\Upsilon\texttt{[m\_ },\kappa\texttt{\_},\{\texttt{kx\_, ky\_, kz\_}\}]\ /;\ (\texttt{m}\ ^2 == 1) := \frac{2\,i}{\sqrt{2\pi}\,\kappa}\texttt{SolidHarmonicS}\,[1, \texttt{m}\ , \texttt{kx, ky, kz}]\,\frac{1}{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}\left(\kappa-\frac{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2+\kappa^2}{\sqrt{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}}\texttt{ArcTan}\left[\sqrt{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}\Big/\kappa\right]\right)$$

$$\texttt{(*equal to }(\texttt{kx}^2\texttt{+ky}^2\texttt{+kz}^2\texttt{+}\kappa^2)\texttt{ times shortRange}\Psi\texttt{.*)}$$

$$\texttt{reduced}\Upsilon\texttt{[m\_ },\kappa\texttt{\_},\{\texttt{kx\_, ky\_, kz\_}\}]\ /;\ (\texttt{m}\ ^2 == 1) := \texttt{SolidHarmonicS}\,[1, \texttt{m}\ , \texttt{kx, ky, kz}]$$

$$\texttt{doubleReduced}\Upsilon\texttt{[m\_ },\kappa\texttt{\_},\{\texttt{kx\_, ky\_, kz\_}\}]\ /;\ (\texttt{m}\ ^2 == 1) := \frac{\texttt{SolidHarmonicS}\,[1, \texttt{m}\ , \texttt{kx, ky, kz}]}{\sqrt{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}}$$

$$\texttt{shortRange}\Psi\left[0,\kappa\_,\{\texttt{kx\_},\texttt{ky\_},\texttt{kz\_}\}\right]:=\frac{1/\kappa}{\sqrt{2}\,\pi}\frac{1}{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2+\kappa^2}$$

$$\texttt{shortRange}\Upsilon\left[0,\kappa\_,\{\texttt{kx\_},\texttt{ky\_},\texttt{kz\_}\}\right]:=\frac{1/\kappa}{\sqrt{2}\,\pi}1\,\texttt{(*equal to }\left(\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2+\kappa^2\right)\texttt{ times shortRange}\Psi\texttt{.*)}$$

$$\texttt{RadialF}\left[0,1,\kappa\_,\{\texttt{kx\_},\texttt{ky\_},\texttt{kz\_}\}\right]:=\frac{2}{\left(\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2+\kappa^2\right)^2}$$

$$\texttt{RadialF}\left[1,1,\kappa\_,\{\texttt{kx\_},\texttt{ky\_},\texttt{kz\_}\}\right]:=\frac{2/\kappa}{\left(\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2+\kappa^2\right)^2}$$

$$\texttt{RadialF}\left[2,1,\kappa\_,\{\texttt{kx\_},\texttt{ky\_},\texttt{kz\_}\}\right]:=\frac{1}{\left(\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2\right)^2}\left(-\frac{5\left(\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2\right)+3\kappa^2}{\left(\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2+\kappa^2\right)^2}+\frac{3}{\kappa\sqrt{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}}\texttt{ArcTan}\left[\frac{\sqrt{\texttt{kx}^2+\texttt{ky}^2+\texttt{kz}^2}}{\kappa}\right]\right)$$

$$\texttt{ConstantN}\left[\texttt{l\_},\texttt{ll\_},\texttt{m\_},\texttt{q\_}\right]:=2^{3/2}\,i^{\texttt{ll}}\,(-1)^{\texttt{q}}\sqrt{\frac{(2\,\texttt{l}+1)\,(2\,\texttt{ll}+1)}{4\pi}}\;\texttt{ThreeJSymbol}\left[\{\texttt{l},0\},\{\texttt{ll},0\},\{1,0\}\right]\texttt{ThreeJSymbol}\left[\{\texttt{l},-\texttt{m}\},\{\texttt{ll},\texttt{m}-\texttt{q}\},\{1,\texttt{q}\}\right]$$

$$\texttt{UnitE[0]} := \{0,0,1\}$$

$$\texttt{UnitE}\left[\texttt{q\_}\,/;\,\left(\texttt{q}^2==1\right)\right]:=\frac{-\texttt{q}}{\sqrt{2}}\{1,\texttt{q}\,i,0\}$$

```
Table[
    shortRangeDipoleConj[m ][{px_, py_, pz_}, κ_] = (0
            +UnitE[+m ]*ConstantN[1, 0, m , m ]
              SolidHarmonicS [0, 0, {px, py, pz}] RadialF[0, 1, κ, {px, py, pz}]
            +UnitE[+m ]*ConstantN[1, 2, m , m ] SolidHarmonicS [2, 0, {px, py, pz}]
              RadialF[2, 1, κ, {px, py, pz}]
            +UnitE[-m ]*ConstantN[1, 2, m , -m ] SolidHarmonicS [2, -2m , {px, py, pz}]
              RadialF[2, 1, κ, {px, py, pz}]
            +UnitE[+0]*ConstantN[1, 2, m , 0] SolidHarmonicS [2, -m , {px, py, pz}]
              RadialF[2, 1, κ, {px, py, pz}]
        ) /. {(px^2 + py^2 + pz^2)^(5/2) → (px^2 + py^2 + pz^2)^2 Sqrt[px^2 + py^2 + pz^2]}
    , {m , {1, -1}}];
```

$$\texttt{shortRangeDipoleConj[0][\{px\_, py\_, pz\_\}},\kappa\_]:=\frac{i\sqrt{2}}{\pi\kappa}\frac{\{\texttt{px},\texttt{py},\texttt{pz}\}}{\left(\texttt{px}^2+\texttt{py}^2+\texttt{pz}^2+\kappa^2\right)^2};$$

Building the dipole

Note that the harmonic-dipole functions encode into the fictional $m = 2$ orbital the full-2p response, as it simplifies a bunch of data-handling structures.

```
rotatingFrameHarmonicDipole[S_, mm_, Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm},
    {t, τ} = saddleAssociation[m];
    {t0, τ0} = saddleAssociation[0];

    i (2π/(i τ))^(3/2) HessianRoot[S, t, τ] ×
      shortRangeDipoleConj[m][pi[Rzα[-τ].ps[t, t-τ], t, t-τ], κ] ×
      shortRangeΥ[m, κ, pi[ps[t0, t0-τ0], t0-τ0, t0-τ0]] × Exp[-i S[t, t-τ] + i Ω t]
  ]

rotatingFrameHarmonicDipoleNoIntrinsicPhase[S_, mm_, Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm},
    {t, τ} = saddleAssociation[m];
    {t0, τ0} = saddleAssociation[0];

    i (2π/(i τ))^(3/2) HessianRoot[S, t, τ] ×
      shortRangeDipoleConj[m][pi[Rzα[-τ].ps[t, t-τ], t, t-τ], κ] ×
      shortRangeΥ[m, κ, pi[ps[t0, t0-τ0], t0-τ0, t0-τ0]] ×
      Abs[Exp[-i S[t, t-τ] + i Ω t]]
  ]
```

Functions for deeper analysis

```
recombinationDipole[S_, mm_, Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm},
    {t, τ} = saddleAssociation[m];
    {t0, τ0} = saddleAssociation[0];

    shortRangeDipoleConj[m][pi[Rzα[-τ].ps[t, t-τ], t, t-τ], κ]
  ]

ionizationFactor[S_, mm_, Ω_, saddleAssociation_] := Block[{t, τ, t0, τ0, m = mm},
    {t, τ} = saddleAssociation[m];
    {t0, τ0} = saddleAssociation[0];

    shortRangeΥ[m, κ, pi[ps[t0, t0-τ0], t0-τ0, t0-τ0]]
  ]

ionizationFactorTotal[S_, mm_, Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm},
    {t, τ} = saddleAssociation[m];
    {t0, τ0} = saddleAssociation[0];

    shortRangeΥ[m, κ, pi[ps[t0, t0-τ0], t0-τ0, t0-τ0]] Abs[Exp[-i S[t, t-τ] + i Ω t]]
  ]
```

```
actionWithoutIpModification[S_, mm_ , Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = 0},
    {t, τ} = saddleAssociation[m];
    Abs[Exp[-ⅈ S[t, t - τ] + ⅈ Ω t]]
  ]

actionWithIpModification[S_, mm_ , Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm },
    {t, τ} = saddleAssociation[m];
    Abs[Exp[-ⅈ S[t, t - τ] + ⅈ Ω t]]
  ]
```

Trajectory functions

```
rotatingFrameTrajectory[ttr_, {t_, tt_}] =
    Rzα[-(ttr - tt)].Integrate[ps[t, tt] + Rzα[-tt].bicircularA[td], {td, tt, ttr}];

rotatingFrameVelocity[ttr_, {t_, tt_}] :=
  Rzα[-(ttr - tt)].ps[t, tt] + bicircularArot[ttr]
```

Functions for analysis

For more details, see https://physics.stackexchange.com/questions/308009/how-can-i-get-the-axes-of-the-polarization-ellipse-from-the-jones-vector-of-the

```
getEllipticity::nnzez = "getEllipticity called with nonzero third argument `1`.";
getEllipticity[{Ex_, Ey_, Ez_}] := (If[N[Ez] ≠ 0., Message[getEllipticity::nnzez, Ez]];
    getEllipticity[{Ex, Ey}])
getEllipticity[{Ex_, Ey_}] := If[
    Total[{Ex, Ey}^2] == 0,
    Sign[Cross[Re[#], Im[#]].{0, 0, 1}] &[{Ex, Ey, 0}]
    ,
    (Sign[Cross[Re[#], Im[#]].{0, 0, 1}] Norm[Im[#]]/Norm[Re[#]] &[
        Sqrt[Total[{Ex, Ey}^*2]]/Abs[Sqrt[Total[{Ex, Ey}^*2]]] {Ex, Ey, 0}])
  ]


getAngle::nnzez = "getAngle called with nonzero third argument `1`.";
getAngle[{Ex_, Ey_, Ez_}] := (If[N[Ez] ≠ 0., Message[getAngle::nnzez, Ez]];
    getAngle[{Ex, Ey}])
getAngle[{Ex_, Ey_}] := Mod[(ArcTan @@ Re[#] &[Sqrt[Total[{Ex, Ey}^*2]]/Abs[Sqrt[Total[{Ex, Ey}^*2]]] {Ex, Ey}]), π]
```

```
getPhaseGamma::nnzez =
    "getPhaseGamma called with nonzero third argument `1`.";
getPhaseGamma[{Ex_, Ey_, Ez_}] :=
  (If[N[Ez] ≠ 0., Message[getPhaseGamma::nnzez, Ez]];
    getPhaseGamma[{Ex, Ey}])
getPhaseGamma[{Ex_, Ey_}] := If[
    Total[{Ex, Ey}^2] == 0,
    Arg[UnitE[getEllipticity[{Ex, Ey}]]*.{Ex, Ey, 0}]
    ,
    1/2 Arg[Total[{Ex, Ey}^2]]
  ]
```

Calculation of the saddle points

Parameters

```
parameters = {getIonizationPotential["Neon", 0], √1.88 0.053, 0.057}; (*{Ip,F,ω}*)
```

Neon, 1.88×10^{14} W/cm ^{2}, 800 nm

Getting the saddle points

Calculation

```
Block[{Ip, F, ω, γ},
  {Ip, F, ω} = parameters ;
  γ = Sqrt[2 Ip ω]/F;
  ΩRange = Range[7 ω, 75 ω, 1/24. ω];

  AbsoluteTiming [
    saddlePoints = Association[Table[
          m  → GetSaddlePoints[
              ΩRange, S, Table[
                {{(0 - ⅈ 2 γ)/ω, (200 ° + ⅈ 2 γ)/ω}, {(55. ° + 0.35 ⅈ)/ω + (120 ° k)/ω, (100 ° + 1.1 ⅈ)/ω + (120 ° k)/ω}}
                , {k, 0, 5}]
              , IndependentVariables → {"τ", "tt"}
              , Tolerance → 10^-5/ω, Seeds → 75
              , Jacobian → FiniteDifference
            ]
          , {m , -1, 1}]]
  ]
]
```

```
{1036.061170,
  <|-1 → <|0.399 → {{176.022+3.10788 ⅈ, 5.43574-11.594 ⅈ}, {102.535+3.10788 ⅈ,
          5.43574-11.594 ⅈ}, {29.0471+3.10788 ⅈ, 5.43574-11.594 ⅈ}, … 6 …,
        {46.8644-11.4101 ⅈ, 21.0255-24.3058 ⅈ}, {120.352-11.4101 ⅈ,
          21.0255-24.3058 ⅈ}, {230.583-11.4101 ⅈ, 21.0255-24.3058 ⅈ}},
      … 1631 …, 4.275 → {{ … 1 … }, … 16 …, { … 1 … }}|>,
    0 → … 1 …, 1 → <| … 1 … |>|>}
```

large output | show less | show more | show all | set size limit ...

Data handling

In-notebook save:

```
With[{data = Compress [saddlePoints]},
  Button["Restore saddle points", Set[saddlePoints, Uncompress [data]];
    saddlePoints;]
]
```

Restore saddle points

External export:

```
Save[NotebookDirectory[] <>
    "data - saddle points on the rotating frame .txt", saddlePoints]
```

Import from external export:

```
<< (NotebookDirectory[] <> "data - saddle points on the rotating frame .txt");
```

Check the import worked correctly:

```
saddlePoints // Dimensions
```

```
{3, 1633}
```

Initial map of the saddle points

```
Block[{Ip, F, ω, γ, saddles},
  {Ip, F, ω} = parameters ;
  γ = Sqrt[2 Ip ω]/F;
  Row[Table[
      saddles = saddlePoints[m ];

      Column [Table[
          Show[
            Graphics[
              ParallelTable[
                Map[
                  Apply[Function[{t, τ},
                      Tooltip[Point[ReIm [ω (time /. {"tt" → t - τ, "t" → t, "τ" → τ})]],
                        {Ω/ω, ω{t - τ, t, τ}, Floor[ωRe[t - τ], 2π/3]/(2π/3)}]
                  ]], saddles[Ω]〚All〛]
                , {Ω, Keys[saddles]〚1 ;; ;; 10〛}]
            ]
            , Frame → True, Axes → True
            , ImageSize → 750
            , FrameLabel → {"Re(ω"<>time <>")", "Im (ω"<>time <>")"}
          ]
          , {time , {"tt", "t", "τ"}}]]
      , {m , {1, -1}}]]
]
```

Classifying the saddle points

Note that the saddle-point classification involves the manual fixing of the parameters ωt_{min} = 0.95 and ωt_{min} = 2.68, chosen so that a vertical strip in τ with those limits will include only the short trajectories. If the parameters change, these might need to be adjusted.

Naive sort

```
Block[{Ip, F, ω, γ, saddles, classifierFunction, sortingFunction, selection},
  {Ip, F, ω} = parameters ;
  γ = Sqrt[2 Ip] ω / F;
  classifierFunction = Function[{t, τ, Ω}, Which@@Flatten[{Table[
            {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π/3] / (2 π/3) == k - 1], "A" <> ToString[k]
            , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];

  Show[
    Table[
      selection = ClassifyQuantumOrbits [saddlePoints[m ],
          classifierFunction, sortingFunction, DiscardedLabels → {"Discard"}];
      Graphics[{
          RGBColor[(1 + m)/2, 0, (1 - m)/2],
          Table[ParallelTable[
              Point[
                ReIm  [ω #2 &@@ selection[index, Ω, 1]]
              ]
              , {Ω, Keys[selection[index]][[1 ;; ;; 1]]}], {index, Keys[selection][[{1}]]}]
        }]
      , {m , -1, 1}]
    , Frame  → True, Axes → True
    , ImageSize  → 600
    , FrameLabel  → {"Re(ωτ)", "Im  (ωτ)"}
  ]

]
```

Refactored sort

This puts the orbital number m as the innermost index so that the $m = 0$ saddles can be used in the

```
Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction, selection},
  {Ip, F, ω} = parameters ; κ = √(2 Ip) ; γ = κω/F;
  classifierFunction= Function[{t, τ, Ω}, Which@@Flatten[{Table[
            {And[0.95 < ωRe[τ] < 2.68, Floor[ωRe[t-τ], 2π/3]/(2π/3) == k-1], "A" <> ToString[k]
            , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction=
    Function[list, SortBy[list, Function[Re[ω#[[1]] - Floor[ωRe[#[[1]] - #[[2]]], 2π]]]]];
  selection= Query[Transpose] /@Query[Transpose] [Association[ParallelTable[
          m  → KeySort[ClassifyQuantumOrbits [saddlePoints[m ], classifierFunction,
               sortingFunction, DiscardedLabels→ {"Discard"}]]
          , {m , -1, 1}]]];

  Column [Table[
      Show[
        Graphics[
          Table[
            Table[{
                Blend[{Hue[ToExpression[StringTake[index, {2}]]/6],
                    GrayLevel[(1+m)/2]}, 0.5],
                Tooltip[
                  Line[
                    Table[
                      Function[{t, τ},
                          ReIm [ω (time /. {"tt" → t-τ, "t" → t, "τ" → τ})]
                        ] @@selection[index, Ω, m , 1]
                      , {Ω, Keys[selection[index]]}]
                  ]
                  , {index, m }]
            }, {m , -1, 1}]
          , {index, Keys[selection]}]
        ]
        , Frame  → True, Axes → True
        , ImageSize  → 900
        , FrameLabel  → {"Re(ω" <> time <> ")", "Im (ω" <> time <> ")"}
      ]


      , {time , {"tt", "t", "τ"}}]]

]
```

Harmonic dipoles and spectra

Calculation of dipole data

Bare calculation

In the dipole data, $m = 2$ corresponds to the full-2p response for simpler data handling.

```
Block[{Ip, F, ω, κ, γ, classifierFunction, sortingFunction},
  {Ip, F, ω} = parameters ; κ = Sqrt[2 Ip] ; γ = κ ω / F;
  classifierFunction = Function[{t, τ, Ω}, Which@@Flatten[{Table[
            {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π / 3] / (2 π / 3) == k - 1], "A" <> ToString[k]}
            , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];

  Print[AbsoluteTiming[
      selection = Query[Transpose] /@ Query[Transpose][Association[ParallelTable[
                m → KeySort[ClassifyQuantumOrbits[saddlePoints[m],
                      classifierFunction, sortingFunction,
                      DiscardedLabels → {"Discard"}]][[All, All, 1]]
                , {m, -1, 1}]]];
    ]];

  Print[DateString[]];
  Print[AbsoluteTiming[
      Dimensions /@ {
          SPAdipoleData = Map[Total,
              Query[Transpose] /@ Association[Table[
                  m → Association[ParallelTable[
                        index → Association[Table[
                              Ω → rotatingFrameHarmonicDipole[
                                  S, m, Ω, selection[index, Ω]]
                              , {Ω, Keys[selection[index]][[1 ;; -1]]}]]
                        , {index, Keys[selection[[1 ;; -1]]]}]]
                  , {m, -1, 1}]]
              , {2}],
          Print["Done with data 1 at ", DateString[]];
          singleSPAdipoleData = Map[Total,
              Query[Transpose] /@ Association[Table[
                  m → Association[ParallelTable[
                        index → Association[Table[
                              Ω → rotatingFrameHarmonicDipole[
                                  S, m, Ω, selection[index, Ω]]
                              , {Ω, Keys[selection[index]][[1 ;; -1]]}]]
                        , {index, Keys[selection[[1 ;; 1]]]}]]
```

```
                , {m , -1, 1}]]
            , {2}],
        Print["Done with data 2 at ", DateString[]];
        singleSPAdipoleDataNoIntrinsicPhase = Map[Total,
            Query[Transpose] /@Association[Table[
                m → Association[ParallelTable[
                    index→ Association[Table[
                        Ω → rotatingFrameHarmonicDipoleNoIntrinsicPhase[
                            S, m , Ω, selection[index, Ω]]
                        , {Ω, Keys[selection[index]][[1 ;; -1]]}]]
                    , {index, Keys[selection[[1 ;; 1]]]}]]
                , {m , -1, 1}]]
            , {2}]
        }
    ]];
  Print["Done with data 3 at ", DateString[]];
  SPAdipoleData[2] = SPAdipoleData[1] + SPAdipoleData[-1];
  singleSPAdipoleData[2] = singleSPAdipoleData[1] + singleSPAdipoleData[-1];
  singleSPAdipoleDataNoIntrinsicPhase[2] =
    singleSPAdipoleDataNoIntrinsicPhase[1] + singleSPAdipoleDataNoIntrinsicPhase[-1];
]
Print[DateString[]];
```

```
{4.013075, Null}

Fri 21 Apr 2017 22:26:22

Done with data 1 at Fri 21 Apr 2017 22:30:17

Done with data 2 at Fri 21 Apr 2017 22:32:42

{522.983914, {{3, 1633}, {3, 1633}, {3, 1633}}}

Done with data 3 at Fri 21 Apr 2017 22:35:05

Fri 21 Apr 2017 22:35:05
```

Data handling

Data handling

In-notebook save:

```
With[{data = Compress [SPAdipoleData]},
  Button["Restore SPA dipole data", Set[SPAdipoleData, Uncompress [data]];
    SPAdipoleData;]
]
With[{data = Compress [singleSPAdipoleData]},
  Button["Restore single SPA dipole data",
    Set[singleSPAdipoleData, Uncompress [data]];
    singleSPAdipoleData;]
]
With[{data = Compress [singleSPAdipoleDataNoIntrinsicPhase]},
  Button["Restore single SPA dipole data without intrinsicphase",
    Set[singleSPAdipoleDataNoIntrinsicPhase, Uncompress [data]];
    singleSPAdipoleDataNoIntrinsicPhase]
]
```

Restore SPA dipole data

Restore single SPA dipole data

Restore single SPA dipole data without intrinsic phase

External export:

```
Save[NotebookDirectory[] <> "data - SPA dipole data on the rotating frame .txt",
  SPAdipoleData]
Save[NotebookDirectory[] <>
    "data - SPA dipole data for a single ionizationburst on the
        rotating frame .txt", singleSPAdipoleData]
Save[NotebookDirectory[] <> "data - SPA dipole data for a single burst,
        with no intrinsicphase, on the rotating frame .txt",
  singleSPAdipoleDataNoIntrinsicPhase]
```

Import from external export:

```
<< (NotebookDirectory[] <> "data - SPA dipole data on the rotating frame .txt");
<< (NotebookDirectory[] <>
        "data - SPA dipole data for a single ionizationburst on the
            rotating frame .txt");
<< (NotebookDirectory[] <> "data - SPA dipole data for a single burst, with
            no intrinsicphase, on the rotating frame .txt");
```

Check the import worked correctly:

```
SPAdipoleData // Dimensions
singleSPAdipoleData // Dimensions
singleSPAdipoleDataNoIntrinsicPhase // Dimensions
```

```
{4, 1633}

{4, 1633}

{4, 1633}
```

Spectra

```
sizeD = 170;

Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction},
  {Ip, F, ω} = parameters ; κ = Sqrt[2 Ip] ; γ = κ ω / F;
  classifierFunction = Function[{t, τ, Ω}, Which @@ Flatten[{Table[
          {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π / 3] / (2 π / 3) == k - 1], "A" <> ToString[k]}
          , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];
  (*selection=Query[Transpose]/@Query[Transpose][Association[ParallelTable[
          m →
            KeySort[ClassifyQuantumOrbits[saddlePoints[m ],classifierFunction,
                sortingFunction, DiscardedLabels→{"Discard"}]][[All,All,1]]
          ,{m ,-1,1}]]];*)


  Column @AbsoluteTiming @Column [Table[Column [{
          figureD[m ] = Show[
              Join[{
                  Graphics[{
                      GrayLevel[0.5],
                      Thickness[0.0025],
                      Tooltip[
                        Line[
                          DeleteCases[
                            ParallelTable[
                              {Ω / ω, Log10[Norm [
                                      SPAdipoleData[m , Ω]
                                      ]^2]}
                              , {Ω, Select[Keys[SPAdipoleData[m ]][[1 ;; ;; 1]],
                                      12 ω ≤ # ≤ 60 ω &]}]
                            , {_, _? (# < -28 &)}]
                        ]
                        , {m }]
                  }],
                  Graphics[{
                      Opacity[0.7],
                      GrayLevel[0.5],
                      Thickness[0.003],
                      Tooltip[
```

```
Line[
  ParallelTable[
    {Ω/ω, Log10[Norm [
          (6 singleSPAdipoleData[m , Ω])
          ]^2]}
    , {Ω, Select[Keys[singleSPAdipoleData[m ]][[
          1 ;; ;; 1]], 12 ω ≤ # ≤ 60 ω &]}]
  ]
  , {m }]
}]
},
Transpose[Table[{
      Graphics[{
          RGBColor[s, 0, 1-s],
          Thickness[0.0025],
          Tooltip[
            Line[
              DeleteCases[
                ParallelTable[
                  {Ω/ω, Log10[Abs[
                  ({1, -s i, 0}/√2).SPAdipoleData[m , Ω]
                  ]^2]}
                  , {Ω, Select[Keys[SPAdipoleData[m ]][[
                  1 ;; ;; 1]], 12 ω ≤ # ≤ 60 ω &]}]
                , {_, _? (# < -28 &)}]
            ]
            , {m , s}]
        }]
      ,
      Graphics[{
          Opacity[0.7],
          RGBColor[s, 0, 1-s],
          Thickness[0.003],
          Tooltip[
            Line[
              ParallelTable[
                {Ω/ω, Log10[Abs[
```

```
                    {1, -s i, 0}
                    ------------ . (6 singleSPAdipoleData[m , Ω])
                        √2
                    ]^2]}
                  , {Ω, Select[Keys[singleSPAdipoleData[m ]][[
                    1 ;; ;; 1]], 12 ω ≤ # ≤ 60 ω &]}]
                ]
                , {m , s}]
            }]
        }
        , {s, {1, -1}}]], {
      Graphics[{Black, Thickness[0.001],
          Line[{{Ip/ω, -10}, {Ip/ω, -5.6}}]}],
      Graphics[{White, FilledCurve[{{Line[ImageScaled /@
                    {{0, -0.1}, {1, -0.1}, {1, 1}, {0, 1}}]},
              {Line[Scaled/@({{0, 0}, {1, 0}, {1, 1}, {0, 1}})]}}]}]
    }]
  , ImageSize → (Automatic → {sizeD, Automatic })
  (*,ImageSize →If[m ==-1||m ==0,sizeLeftD,sizeRightD]*)
  , AspectRatio→ 1/1.6
  , Frame → True
  , PlotRange → {{12, 60}, {-9.2, -5.6}}
  , PlotRangeClipping→ False
  (*,PlotRangeClipping→True*)
  , FrameTicks → {{If[m == -1 || m == 0, ## &[], None], Join[
          {#, If[m == -1 || m == 0, MaTeX["10^{" <> ToString[# + 6] <>
                        "}" /. {"10^{0}" → "1"}, FontSize→ tfs],
                  ""], {0.01, 0}} & /@Range[-10, -6],
          {#, "", {0.005, 0}} & /@Flatten[Outer[Plus,
                Log10[Range[2., 9.]], Range[-11, -5]]]
        ], If[m == -1 || m == 0, None, ## &[]]}, {Join[
          {#, ""(*MaTeX[ToString[#],FontSize→tfs]*),
                {0.016, 0}} & /@Range[0, 60, 3],
          {#, "", {0.009, 0}} & /@Range[1, 60, 1]
        ], Join[{#, "", {0.016, 0}} & /@Range[0, 60, 3],
          {#, "", {0.009, 0}} & /@Range[1, 60, 1]]}}
  , FrameLabel → {None, If[m == -1 || m == 0,
        MaTeX["|\\mathbf D(\\Omega )|^2\\ \\mathrm {(arb.u.)}",
          FontSize→ lfs], None]}
  , GridLines→ {Range[3/2, 90, 3], Join[
        Range[-12, -6],
        Flatten[
          Outer[Plus, Log10[Range[2., 9.]], Range[-11, -5]]]
      ]}
  , GridLinesStyle→ Directive[GrayLevel[0.8]]
  , ImagePadding → {
      (*{If[m ==-1||m ==0,37,3],3},*)
```

```
{If[m == -1 || m == 0,
    Scaled[0.085], Scaled[0.006]], Scaled[0.006]},
{Scaled[0.001], Scaled[0.001]}
(*{If[m ==0||m ==2,56,30],0.01}*)
}
, Epilog→ {
  Inset[MaTeX["m =" <> ToString[m ] /. {"m =2" → "2p"}, FontSize→ if
    Scaled[{0.5, 0.98}], Scaled[{0.5, 1}]],
  Inset[MaTeX["\\text{(" <> {"a", "c", "b", "d"}[[m +2]] <> ")}",
      FontSize→ ifs], Scaled[{0.04, 0.98}], Scaled[{0, 1}]],
  If[m == 1, Inset[Grid[{
        {Graphics[{Red, Thickness[0.05], Line[{{0, 0}, {1, 0}}]}
            PlotRange → {{0, 1}, 0.1 {-1, 1}},
            PlotRangePadding→ None, ImageSize → 10],
          MaTeX["\\rcirclearrowleft", FontSize→ 5]},
        {Graphics[{Blue, Thickness[0.05], Line[{{0, 0},
                {1, 0}}]}, PlotRange → {{0, 1}, 0.1 {-1, 1}},
            PlotRangePadding→ None, ImageSize → 10],
          MaTeX["\\lcirclearrowright", FontSize→ 5]}
      }, Background → White, Spacings→ {0.5, 0},
      Frame → True, FrameStyle → Directive[
          Gray, Thickness[0.00001]]]
    , Scaled[{0.99, 0.99}], Scaled[{1, 1}]], ## &[]]
}
]

,

figureDellipticity[m ] = Show[{
    Graphics[{Black, Thickness[0.001], Line[
          {{Ip / ω, -1.2}, {Ip / ω, 1.2}}], Line[{{12, 0}, {60, 0}}]}],
    Graphics[{
        Black,
        RGBColor[0.0, 0., 0.8],
        RGBColor[0.025, 0.01, 0.01],
        Thickness[0.003],
        Line[
          ParallelTable[
            {Ω / ω, getEllipticity[
                singleSPAdipoleData[m , Ω][[{1, 2}]]]}
            , {Ω, Select[Keys[singleSPAdipoleData[m ]][[
                  1 ;; ;; 1]], 12 ω ≤ # ≤ 60 ω &]}]
          ]
        }]
  }
  , ImageSize → (Automatic → {sizeD, Automatic })
  , AspectRatio→ 1 / 5
  , Frame → True
  , PlotRange → {{12, 60}, {-1.2, 1.2}}
  , PlotRangeClipping→ True
  , PlotRangePadding→ {{None, None}, {Scaled[0.005], Scaled[0.005]}}
```

```
, GridLines→ {Range[3/2, 90, 3], Range[-1, 1, 0.2]}
, GridLinesStyle→ Directive[GrayLevel[0.8]]
, Axes → False
, Axes → True
, Method → {"AxesInFront" → False}
, AxesOrigin→ {0, 0}
, FrameLabel  → {If[m  == 0 || m  == 2, MaTeX["\\Omega /\\omega ",
        FontSize→ lfs], None], If[m  == -1 || m  == 0,
      MaTeX["\\varepsilon", FontSize→ lfs], None]}
, FrameTicks  → {
    {If[m  == -1 || m  == 0, ## &[], None],
      Join[{#, If[m  == -1 || m  == 0, MaTeX[#, FontSize→ ilfs], ""],
              {0.01, 0}} & /@Range[-1, 1],
        {#, "", {0.005, 0}} & /@Range[-1, 1, 0.2]],
      If[m  == 1 || m  == 2, ## &[], None]
    },
    {Join[
        {#, MaTeX[ToString[#], FontSize→ tfs],
              {0.016, 0}} & /@Range[0, 60, 3],
        {#, "", {0.009, 0}} & /@Range[1, 60, 1]
      ], Join[{#, "", {0.016, 0}} & /@Range[0, 60, 3],
        {#, "", {0.009, 0}} & /@Range[1, 60, 1]]}}

, ImagePadding  → {
    (*{If[m ==-1||m ==0,37,3],3},*)
    {If[m  == -1 || m  == 0,
        Scaled[0.085], Scaled[0.006]], Scaled[0.006]},
    (*{Scaled[0.001],Scaled[0.001]}*)
    {If[m  == 0 || m  == 2,
        Scaled[0.06], Scaled[0.001]], Scaled[0.001]}
    (*{If[m ==0||m ==2,200,200],5}*)
  }
]
}], {m , -1, 2}]]

]
Table[{
    FileByteCount[
      Export[FileNameJoin [{$OutputDirectory, "figureD" <> {"a", "c", "b", "d"}[[m  + 2]] <>
            "-spectra-" <> {"p-", "s", "p+", "2p"}[[m  + 2]] <> ".pdf"}], figureD[m ]]],
    FileByteCount[Export[FileNameJoin [{$OutputDirectory,
          "figureD" <> {"a", "c", "b", "d"}[[m  + 2]] <> "-ellipticity" <>
            {"p-", "s", "p+", "2p"}[[m  + 2]] <> ".pdf"}], figureDellipticity[m ]]]
  }, {m , -1, 2}]
pdflatex[]
{{42 694, 8242}, {41 677, 13 626}, {39 438, 7450}, {37 797, 14 055}}

pdflatex successful
```

Internal structure of the harmonic dipole

Function definitions

```
recombinationDipole[S_, mm_ , Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm },
    {t, τ} = saddleAssociation[m ];
    {t0, τ0} = saddleAssociation[0];

    shortRangeDipoleConj[m ][pi[Rzα[-τ].ps[t, t-τ], t, t-τ], κ]
  ]

ionizationFactor[S_, mm_ , Ω_, saddleAssociation_] := Block[{t, τ, t0, τ0, m = mm },
    {t, τ} = saddleAssociation[m ];
    {t0, τ0} = saddleAssociation[0];

    shortRangeϒ[m , κ, pi[ps[t0, t0-τ0], t0-τ0, t0-τ0]]
  ]

ionizationFactorTotal[S_, mm_ , Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm },
    {t, τ} = saddleAssociation[m ];
    {t0, τ0} = saddleAssociation[0];

    (1/τ)^(3/2) shortRangeϒ[m , κ, pi[ps[t0, t0-τ0], t0-τ0, t0-τ0]]
      Abs[Exp[-ⅈ S[t, t-τ] + ⅈ Ω t]]
  ]

ionizationFactorϒExponential[S_, mm_ , Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm },
    {t, τ} = saddleAssociation[m ];
    {t0, τ0} = saddleAssociation[0];

    shortRangeϒ[m , κ, pi[ps[t0, t0-τ0], t0-τ0, t0-τ0]] × Abs[Exp[-ⅈ m ω/2 (t-τ)]]
  ]

ionizationFactorAction[S_, mm_ , Ω_, saddleAssociation_] :=
  Block[{t, τ, t0, τ0, m = mm },
    {t, τ} = saddleAssociation[m ];
    {t0, τ0} = saddleAssociation[0];

    (1/τ)^(3/2) Abs[Exp[-ⅈ S[t, t-τ] + ⅈ Ω t]]
  ]
```

```
(*rotatingFrameHarmonicDipoleNoIntrinsicPhase[S_,mm_ ,Ω_,saddleAssociation]:=
  Block[{t,τ,t0,τ0,m =mm },
    {t,τ}=saddleAssociation[m ];
    {t0,τ0}=saddleAssociation[0];

    i (2π/(i τ))^(3/2) HessianRoot[S,t,τ]×
      shortRangeDipoleConj[m ][pi[Rzα[-τ].ps[t,t-τ],t,t-τ],κ]×
      shortRangeϒ[m ,κ,pi[ps[t0,t0-τ0],t0-τ0,t0-τ0]]×Abs[Exp[-i S[t,t-τ]+i Ω t]]
  ]*)
```

Figure E

Calculations and data handling

```
Block[{Ip, F, ω, κ, ϒ, saddles, classifierFunction, sortingFunction},
  {Ip, F, ω} = parameters ; κ = Sqrt[2 Ip] ; ϒ = κω/F;
  classifierFunction= Function[{t, τ, Ω}, Which@@Flatten[{Table[
               {And[0.95 < ωRe[τ] < 2.68, Floor[ωRe[t-τ], 2π/3]/(2π/3) == k-1], "A"<>ToString[k]
               , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction=
    Function[list, SortBy[list, Function[Re[ω#[[1]] - Floor[ωRe[#[[1]] - #[[2]]], 2π]]]]];
  Print[AbsoluteTiming [
      selection= Query[Transpose] /@Query[Transpose] [Association[ParallelTable[
                    m  → KeySort[ClassifyQuantumOrbits [saddlePoints[m ],
                            classifierFunction, sortingFunction,
                            DiscardedLabels→ {"Discard"}]][[All, All, 1]]
                    , {m  , -1, 1}]]];
    ]];

  Print[DateString[]];

  Print[AbsoluteTiming [Dimensions [
        internalStructureData= Association[Table[
              Print["Starting ", function, " at ", DateString[]];
              function→ Map[Total,
                  Query[Transpose] /@Association[Table[
                        m  → Association[ParallelTable[
                              index→ Association[Table[
                                    Ω→ function[S, m  , Ω, selection[index, Ω]]
                                    , {Ω, Keys[selection[index]][[1 ;; -1 ;; 1]]}]]
                              , {index, Keys[selection[[1 ;; 1]]]}]]
                        , {m  , -1, 1}]]
```

```
                    , {2}]
                , {function, {recombinationDipole,
                    ionizationFactorϒ, ionizationFactorϒExponential,
                    ionizationFactorTotal, ionizationFactorAction}}]]
        ]]];

    Print[DateString[]];

    Print[AbsoluteTiming [Dimensions [
            figureElineData = Association[Table[
                    Print["Starting ", function, " at ", DateString[]];
                    function → Association[Table[
                            m  → Association[Table[
                                    s → ParallelTable[
                                        {Ω / ω, Log10 [Abs [
                                                If [function === recombinationDipole,
                                                ({1, -s ⅈ, 0} / √2) .
                                                internalStructureData[function, m , Ω],
                                                internalStructureData[function, m , Ω]
                                                ]
                                            ]^2]}
                                        , {Ω, Select [Keys [internalStructureData[function,
                                                m ]][[1 ;; ;; 1]], 12 ω ≤ # ≤ 61 ω &]}]
                                    , {s, {1, -1}}]]
                            , {m , -1, 1}]]
                , {function, {recombinationDipole,
                    ionizationFactorϒ, ionizationFactorϒExponential,
                    ionizationFactorTotal, ionizationFactorAction}}]]
        ]]];

    Print[DateString[]];

]
{6.078559, Null}
Wed 3 May 2017 19:49:15
Starting recombinationDipole  at Wed 3 May 2017 19:49:15
Starting ionizationFactorϒ at Wed 3 May 2017 19:51:36
Starting ionizationFactorϒExponentialat Wed 3 May 2017 19:53:54
Starting ionizationFactorTotal at Wed 3 May 2017 19:56:14
Starting ionizationFactorActionat Wed 3 May 2017 19:58:30
```

```
{691.231193, {5, 3, 1633}}

Wed 3 May 2017 20:00:46

Starting recombinationDipole  at Wed 3 May 2017 20:00:46

Starting ionizationFactorϒat Wed 3 May 2017 20:01:14

Starting ionizationFactorϒExponentialat Wed 3 May 2017 20:01:41

Starting ionizationFactorTotal at Wed 3 May 2017 20:02:08

Starting ionizationFactorActionat Wed 3 May 2017 20:02:36

{137.498135, {5, 3, 2}}

Wed 3 May 2017 20:03:04
```

```
Save[NotebookDirectory[] <> "data - internal structure of the harmonic  dipole.txt",
  internalStructureData]
Save[NotebookDirectory[] <> "data - line data for figure F, internal structure.txt",
  figureElineData]
```

```
<< (NotebookDirectory[] <> "data - internal structure of the harmonic  dipole.txt");
<< (NotebookDirectory[] <> "data - line data for figure F, internal structure.txt");
```

Figure

```
sizeE = 115;
paddingE[mfront_ ] := {If[mfront  == -1, 30, 3.5], 3.5}
labelPositionE = -0.16;

Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction},
  {Ip, F, ω} = parameters  ; κ = Sqrt[2 Ip] ; γ = κω/F;
  classifierFunction = Function[{t, τ, Ω}, Which @@ Flatten[{Table[
          {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π / 3]/(2 π / 3) == k - 1], "A" <> ToString[k]
          , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];


  Row[Table[
      Column  [{


          figureE1[mfront  ] = Show[
              Join[
                Table[
                  If[m  == 1, Reverse, # &] @Table[
                      {
```

```
` Graphics[{
        Thickness[0.006],
        If[m == mfront , RGBColor[
            s, 0, 1-s], GrayLevel[0.7]],
        Tooltip[Line[figureElineData[recombinationDipole,
              m , s]], {m , s}]
      }]
    }
    , {s, {1, -1}}]
  , {m , Join[DeleteCases[Range[-1, 1], mfront ], {mfront }]}], {
  Graphics[{White, FilledCurve[
        {{Line[ImageScaled /@{{0, -0.1}, {1, -0.1}, {1, 1}, {0, 1}}]},
          {Line[Scaled/@ ({{0, 0}, {1, 0}, {1, 1}, {0, 1}})]}}]}],
  Graphics[{Inset[Rotate[MaTeX["|\\mathbf {d}_\\mathrm  {rec}|^2",
          FontSize→ lfs], 90 °],
        Scaled[{labelPositionE, 0.5}], Scaled[{1, 0.5}]]}],
  Graphics[{Inset[MaTeX["m ="<>ToString[mfront ], FontSize→ lfs],
        Scaled[{0.5, 1.005}], Scaled[{0.5, 0}]]}],
  Graphics[{Inset[MaTeX["(\\mathrm  {"<>
          {"a", "b", "c"}[[mfront +2]]<>"})", FontSize→ lfs],
        Scaled[{0, 1}], Scaled[{-0.1, 1.1}]]}]
  }
]
, ImageSize → (Automatic → sizeE)
, AspectRatio→ 1/1.6
, Frame → True
, PlotRange→ {{12, 60}, {-4.9, -1.55}}
, FrameTicks → {{Join[
      {#, MaTeX["10^{"<>ToString[#]<>"}" /. {"10^{0}" → "1"}, FontSize
          {0.01, 0}} &/@Range[-6, 0],
      {#, "", {0.005, 0}} &/@Flatten[Outer[Plus,
          Log10[Range[2., 9.]], Range[-6, 0]]]
    ], None}, {Join[
      {#, "", {0.016, 0}} &/@Range[0, 60, 3],
      {#, "", {0.009, 0}} &/@Range[1, 60, 1]
    ], Join[{#, "", {0.016, 0}} &/@Range[0, 60, 3],
      {#, "", {0.009, 0}} &/@Range[1, 60, 1]]}}
, GridLines→ {Range[3/2, 90, 3], Flatten[Outer[Plus,
      Log10[Range[1., 9.]], Range[-7, 0]]]}
, GridLinesStyle→ Directive[GrayLevel[0.8]]
, ImagePadding → {paddingE[mfront ], {Scaled[0.001], Scaled[0.02]}}
]

,

figureE2[mfront ] = Show[
  Join[
    Table[
      {
        If[m ≠ 0,
```

```
          Graphics[{
              Dashed,
              Thickness[0.006],
              If[m == mfront , GrayLevel[0.3], GrayLevel[0.7]],
              Tooltip[Line[figureElineData[
                    ionizationFactorΥExponential[m , 1]]], {m }]
            }]
          , ## &[]],
        Graphics[{
            Thickness[0.006],
            If[m == mfront , Black, GrayLevel[0.7]],
            Tooltip[
              Line[figureElineData[ionizationFactorΥ[m , 1]]], {m }]
          }]
      }
      , {m , Join[DeleteCases[Range[-1, 1], mfront ], {mfront }]}], {
      Graphics[{White, FilledCurve[
            {{Line[ImageScaled /@{{0, -0.1}, {1, -0.1}, {1, 1}, {0, 1}}]},
              {Line[Scaled/@({{0, 0}, {1, 0}, {1, 1}, {0, 1}})]}}]}],
      Graphics[{Inset[Rotate[MaTeX["|\\Upsilon_\\mathrm {ion}|^2",
                FontSize→lfs], 90 °],
            Scaled[{labelPositionE, 0.5}], Scaled[{1, 0.5}]]}],
      Graphics[{Inset[MaTeX["(\\mathrm {"<>
                {"d", "e", "f"}[[mfront +2]]<>"})", FontSize→lfs],
            Scaled[{0, 1}], Scaled[{-0.1, 1.1}]]}]
    }]
  , ImageSize → (Automatic →sizeE)
  , AspectRatio→1/2
  , Frame →True
  , PlotRange→{{12, 60}, {-1.95, -0.65}}
  , FrameTicks →{{Join[
          {#, "", {0.016, 0}} &/@Range[-2, 1],
          {#, If[EvenQ[Rationalize[100×10^#]],
                MaTeX[PaddedForm [10^#, {3, 2}],
                  FontSize→tfs], ""], {0.008, 0}} &/@
            Flatten[Outer[Plus, Log10[Range[1., 9.]], Range[-2, 1]]]
        ], None}, {Join[
          {#, MaTeX[ToString[#], FontSize→tfs], {0.016, 0}} &/@
            Range[0, 60, 3],
          {#, "", {0.009, 0}} &/@Range[1, 60, 1]
        ], Join[{#, "", {0.016, 0}} &/@Range[0, 60, 3],
          {#, "", {0.009, 0}} &/@Range[1, 60, 1]]}}
  , GridLines→{Range[3/2, 90, 3], Join[
        Range[-2, 1],
        Flatten[Outer[Plus, Log10[Range[2., 9.]], Range[-2, 1]]]
      ]}
  , GridLinesStyle→Directive[GrayLevel[0.8]]
  , ImagePadding →{paddingE[mfront ], {Scaled[0.001], Scaled[0.001]}}
]

,
```

```
figureE3[mfront_] = Show[
    Join[
      Table[
        {
          Graphics[{
              Thickness[0.006],
              If[m == mfront, Black, GrayLevel[0.7]],
              Tooltip[Line[
                  figureElineData[ionizationFactorAction, m, 1]], {m}]
            }]
        }
        , {m, Join[DeleteCases[Range[-1, 1], mfront], {mfront}]}], {
        Graphics[{White, FilledCurve[
              {{Line[ImageScaled /@ {{0, -0.1}, {1, -0.1}, {1, 1}, {0, 1}}]},
                {Line[Scaled /@ ({{0, 0}, {1, 0}, {1, 1}, {0, 1}})]}}]}],
        Graphics[{Inset[Rotate[MaTeX["|e^{-iS}|^2", FontSize -> lfs], 90 °],
              Scaled[{labelPositionE, 0.5}], Scaled[{1, 0.5}]]}],
        Graphics[{Inset[MaTeX["(\\mathrm{" <>
                  {"g", "h", "i"}[[mfront + 2]] <> "})", FontSize -> lfs],
              Scaled[{0, 1}], Scaled[{-0.1, 1.1}]]}]
      }]
    , ImageSize -> (Automatic -> sizeE)
    , AspectRatio -> 1/3
    , Frame -> True
    , PlotRange -> {{12, 60}, {-10.9, -8.9}}
    , FrameTicks -> {{Join[
            {#, "", {0.016, 0}} & /@ Range[-11, -9],
            {#, If[MemberQ[{1, 2, 5}, Round[10 MantissaExponent[
                          10^(#+9)][[1]]]], MaTeX[PaddedForm[10^(#+9),
                        {3, 2}], FontSize -> tfs], ""], {0.008, 0}} & /@
              Flatten[Outer[Plus, Log10[Range[1., 9.]], Range[-11, -9]]]
          ], None}, {Join[
            {#, MaTeX[ToString[#], FontSize -> tfs], {0.016, 0}} & /@
              Range[0, 60, 3],
            {#, "", {0.009, 0}} & /@ Range[1, 60, 1]
          ], Join[{#, "", {0.016, 0}} & /@ Range[0, 60, 3],
            {#, "", {0.009, 0}} & /@ Range[1, 60, 1]]}}
    , GridLines -> {Range[3/2, 90, 3], Flatten[Outer[Plus,
            Log10[Range[1., 9.]], Range[-11, -6]]]}
    , GridLinesStyle -> Directive[GrayLevel[0.8]]
    , ImagePadding -> {paddingE[mfront], {Scaled[0.001], Scaled[0.001]}}
  ]

,

figureE4[mfront_] = Show[
    Join[
      Table[
        {
```

```
                    ` Graphics[{
                        Thickness[0.006],
                        If[m == mfront , Black, GrayLevel[0.7]],
                        Tooltip[
                          Line[figureElineData[ionizationFactorTotal, m , 1]], {m }]
                      }]
                    }
                    , {m , Join[DeleteCases[Range[-1, 1], mfront ], {mfront }]}], {
                    Graphics[{White, FilledCurve[
                          {{Line[ImageScaled /@{{0, -0.1}, {1, -0.1}, {1, 1}, {0, 1}}]},
                            {Line[Scaled/@({{0, 0}, {1, 0}, {1, 1}, {0, 1}})]}}]}],
                    Graphics[{Inset[MaTeX["\\Omega /\\omega ", FontSize→ lfs],
                          Scaled[{0.5, -0.30}], Scaled[{0.5, 1}]]}],
                    Graphics[{Inset[Rotate[MaTeX["|\\Upsilon_\\mathrm {ion}
                                e^{-iS}|^2", FontSize→ lfs], 90 °],
                          Scaled[{labelPositionE, 0.5}], Scaled[{1, 0.5}]]}],
                    Graphics[{Inset[MaTeX["(\\mathrm {"<>
                              {"j", "k", "l"}[[mfront +2]]<>"})", FontSize→ lfs],
                          Scaled[{0, 1}], Scaled[{-0.1, 1.1}]]}]
                  }]
                , ImageSize → (Automatic → sizeE)
                , AspectRatio→ 1/3
                , Frame → True
                , PlotRange→ {{12, 60}, {-11.45, -10.7}}
                , FrameTicks → {{Join[
                        {Log10[1.6] +#, "", {0.016, 0}} &/@Range[-12, 1],
                        {Log10[1.6] +#, If[MemberQ [{1, 2, 4, 6, 8},
                                Round[10 MantissaExponent[10^(#+11)][[1]]]],
                              MaTeX[PaddedForm [10^(#+11), {2, 1}],
                                FontSize→ tfs], ""], {0.008, 0}} &/@
                          Flatten[Outer[Plus, Log10[Range[1., 9.]], Range[-12, -10]]]
                      ], None}, {Join[
                        {#, MaTeX[ToString[#], FontSize→ tfs], {0.016, 0}} &/@
                          Range[0, 60, 3],
                        {#, "", {0.009, 0}} &/@Range[1, 60, 1]
                      ], Join[{#, "", {0.016, 0}} &/@Range[0, 60, 3],
                        {#, "", {0.009, 0}} &/@Range[1, 60, 1]]}}
                , GridLines→ {Range[3/2, 90, 3], Log10[1.6] +
                      Flatten[Outer[Plus, Log10[Range[1., 9.]], Range[-13, -6]]]}
                , GridLinesStyle→ Directive[GrayLevel[0.8]]
                , ImagePadding → {paddingE[mfront ], {Scaled[0.05], Scaled[0.001]}}
              ]

          }]
      , {mfront , -1, 1}]]

]
Table[FileByteCount[Export[FileNameJoin [{$OutputDirectory,
          "figureE"<>{"a", "b", "c"}[[m +2]]<>"-recombination-dipole-"<>
```

```
            {"p-", "s", "p+"}[[m +2]] <> ".pdf"}], figureE1[m ]]], {m , -1, 1}]
Table[FileByteCount[Export[FileNameJoin[{$OutputDirectory,
          "figureE" <> {"d", "e", "f"}[[m +2]] <> "-ionizationfactor-" <>
            {"p-", "s", "p+"}[[m +2]] <> ".pdf"}], figureE2[m ]]], {m , -1, 1}]
Table[FileByteCount[Export[FileNameJoin[{$OutputDirectory,
          "figureE" <> {"g", "h", "i"}[[m +2]] <> "-action-" <>
            {"p-", "s", "p+"}[[m +2]] <> ".pdf"}], figureE3[m ]]], {m , -1, 1}]
Table[FileByteCount[Export[FileNameJoin[{$OutputDirectory,
          "figureE" <> {"j", "k", "l"}[[m +2]] <> "-total-" <>
            {"p-", "s", "p+"}[[m +2]] <> ".pdf"}], figureE4[m ]]], {m , -1, 1}]
pdflatex[]
{43807, 40681, 40710}

{31199, 27262, 27300}

{25100, 20995, 20799}

{32423, 29110, 28857}

pdflatex successful
```

Ellipsometry

Figure F - in 2D

```
Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction},
  {Ip, F, ω} = parameters ; κ = Sqrt[2 Ip] ; γ = κ ω / F;
  classifierFunction = Function[{t, τ, Ω}, Which @@ Flatten[{Table[
        {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π / 3] / (2 π / 3) == k - 1], "A" <> ToString[k]
        , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];
  selection = Query[Transpose] /@ Query[Transpose] [Association[ParallelTable[
          m → KeySort[ClassifyQuantumOrbits [saddlePoints[m ], classifierFunction,
                sortingFunction, DiscardedLabels → {"Discard"}]][[All, All, 1]]
          , {m , -1, 1}]]];

  Column @AbsoluteTiming [Table[

        figureF[m ] = Show[
            {
              Table[
                ListPlot[
                    Tooltip[
                      Table[
```

```
            Re[e^(-i ω t) singleSPAdipoleDataNoIntrinsicPhase[m , Ω]〚{1, 2}〛]
            , {ωt, 10 °, 360 °, 1. °}]
          , {m , Ω/ω}]
        , Joined→ True
        , PlotStyle→
          Directive[Thickness[0.003], ColorData["Rainbow"][1 - (Ω/ω-12)/(60-12)]]
      ] /. {Line[pts_] :→ {If[m ≠ 0, Arrowheads[0.025],
              Arrowheads[0.015]], Line[pts], Arrow[pts]}}
    , {Ω, Select[Keys[singleSPAdipoleData[1]]〚13 ;; ;; 72〛,
        12 ω < # ≤ 61.5 ω &]}],
  Graphics[{White, FilledCurve[
        {{Line[ImageScaled /@ {{0, -0.1}, {1, -0.1}, {1, 1}, {0, 1}}]},
          {Line[Scaled /@ ({{0, 0}, {1, 0}, {1, 1}, {0, 1}})]}}]}],
  Graphics[Inset[MaTeX["\\mathrm  {(" <> {"a", "c", "b", "d"}〚m +2〛 <>
          ")}\\   " <> ("m =" <> ToString[m ] /. {"m =2" → "2p"}),
        FontSize→ ifs], Scaled[{0.03, 0.985}], Scaled[{0, 1}]]],
  Graphics[{If[m == -1,
        Inset[MaTeX["\\Omega /\\omega ", FontSize→ 5],
          {-0.0000815, -0.0001 + 0.000085 + 0.000007}, Scaled[{0, 0.65}]]
      ]}],
  Graphics[{
      If[m == -1,
        Table[{
            {ColorData["Rainbow"][1 - (Ω/ω-12)/(60-12)],
              Thickness[0.003], Arrowheads[0.015],
              Arrow[{{-0.0001, -0.0001 + 0.000085 (1 - (Ω/ω-12)/(61.5-12))}, {
                    If[Divisible[Round[Ω/ω - 1.5], 6], -0.000082,
                      -0.000084], -0.0001 + 0.000085 (1 - (Ω/ω-12)/(61.5-12))}}]},
            {If[Divisible[Round[Ω/ω - 1.5], 6], Inset[MaTeX[Ω/ω,
                    FontSize→ 5], {-0.0000815, -0.0001 +
                      0.000085 (1 - (Ω/ω-12)/(61.5-12))}, Scaled[{0, 0.65}]]]}
          }, {Ω, Select[Keys[singleSPAdipoleData[1]]〚13 ;; ;; 72〛,
              12 ω < # ≤ 61.5 ω &]}]
      ]
    }]

}
, ImageSize → (Automatic → 170)
, Frame → True
, PlotRange → 0.000107 {{-1, 1}, {-1, 1}}
```

```
            , PlotRangePadding→None
            , PlotRange→All
            , AspectRatio→Automatic
            , AxesStyle→GrayLevel[0.7]
            , Method→{"AxesInFront"→False}
            , ImagePadding →{
                If[m == -1 || m == 0, {Scaled[0.08], Scaled[0.001]},
                  {Scaled[0.001], Scaled[0.035]}],
                If[m == 0 || m == 2, {Scaled[0.065], Scaled[0.001]},
                  {Scaled[0.001], Scaled[0.03]}]
              }
            , FrameTicks →{
                If[m == -1 || m == 0,
                  {Join[{0.0001#, MaTeX[PaddedForm [#, {2, 1}], FontSize→tfs],
                            {0.02, 0}} & /@Range[-1, 1, 0.5],
                      {0.0001#, "", {0.01, 0}} & /@Range[-1, 1, 0.1]], None},
                  {None, Join[{0.0001#, MaTeX[PaddedForm [#, {2, 1}], FontSize→tfs],
                            {0.02, 0}} & /@Range[-1, 1, 0.5],
                      {0.0001#, "", {0.01, 0}} & /@Range[-1, 1, 0.1]]}
                ],
                If[m == 0 || m == 2,
                  {Join[{0.0001#, MaTeX[PaddedForm [#, {2, 1}], FontSize→tfs],
                            {0.02, 0}} & /@Range[-1, 1, 0.5],
                      {0.0001#, "", {0.01, 0}} & /@Range[-1, 1, 0.1]], None},
                  {None, Join[{0.0001#, MaTeX[PaddedForm [#, {2, 1}], FontSize→tfs],
                            {0.02, 0}} & /@Range[-1, 1, 0.5],
                      {0.0001#, "", {0.01, 0}} & /@Range[-1, 1, 0.1]]}
                ]
              }
            , FrameLabel →
              {If[m == 0 || m == 2, MaTeX["D_x(\\Omega )\\quad\\text{(arb.u.)}",
                    FontSize→lfs], ""], If[m == -1 || m == 0,
                  MaTeX["D_y(\\Omega )\\text{ (arb.u.)}", FontSize→lfs], ""]}
          ]
        , {m , {-1, 1, 0, 2}}]]
]
AbsoluteTiming [{
    FileByteCount[Export[FileNameJoin [{$OutputDirectory,
            "figureFa-polarization-ellipse-2d-p-.pdf"}], figureF[-1]]],
    FileByteCount[Export[FileNameJoin [{$OutputDirectory,
            "figureFb-polarization-ellipse-2d-p+.pdf"}], figureF[1]]],
    FileByteCount[Export[FileNameJoin [{$OutputDirectory,
            "figureFc-polarization-ellipse-2d-s.pdf"}], figureF[0]]],
    FileByteCount[Export[FileNameJoin [{$OutputDirectory,
            "figureFd-polarization-ellipse-2d-2p.pdf"}], figureF[2]]]}]
pdflatex[]
{2.238941, {56326, 45480, 47308, 47578}}

pdflatex successful
```

```
FileByteCount[Export[FileNameJoin[
      {$OutputDirectory, "figureFa-polarization-ellipse-2d-p-.pdf"}], figureF[-1]]]
FileByteCount[Export[FileNameJoin[{$OutputDirectory,
        "figureFb-polarization-ellipse-2d-p+.pdf"}], figureF[1]]]
FileByteCount[Export[FileNameJoin[{$OutputDirectory,
        "figureFc-polarization-ellipse-2d-s.pdf"}], figureF[0]]]
FileByteCount[Export[FileNameJoin[{$OutputDirectory,
        "figureFd-polarization-ellipse-2d-2p.pdf"}], figureF[2]]]
```

```
80545

80402

76234

81586
```

```
pdflatex[]
```

```
pdflatex successful
```

Figure G - in 3D

In-notebook save:

```
With[{data = Compress[figureGdata]},
  Button["Restore figure G data", Set[figureGdata, Uncompress[data]];
    figureGdata;]
]
```

Restore figure G data

External export:

```
Save[NotebookDirectory[] <> "data - 3D data for ellipsometry.txt", figureGdata]
```

```
Save[NotebookDirectory[] <> "data - figure G uncompressed.txt", figureG]
DumpSave[NotebookDirectory[] <> "data - figure G compressed.mx", figureG]
```

Import from external export:

```
<< (NotebookDirectory[] <> "data - 3D data for ellipsometry.txt");
```

Check the import worked correctly:

```
figureGdata /@ Range[-1, 2] // Dimensions
```

```
{4, 1189, 145, 3}
```

```
Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction},
  {Ip, F, ω} = parameters ; κ = Sqrt[2 Ip] ; γ = κ ω / F;
  classifierFunction = Function[{t, τ, Ω}, Which @@ Flatten[{Table[
        {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π/3] / (2 π/3) == k - 1], "A" <> ToString[k]]
        , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];
  (*selection=Query[Transpose]/@Query[Transpose][Association[ParallelTable[
          m →
            KeySort[ClassifyQuantumOrbits[saddlePoints[m ],classifierFunction,
                  sortingFunction,DiscardedLabels→{"Discard"}]][[All,All,1]]
          ,{m ,-1,1}]]];*)

  Print[DateString[]];

  AbsoluteTiming[
    Dimensions[(*333s≈5m30s  for the whole set at 10° intervals. 217s≈
        3m40s  for a single pop at 2.5°.*)
      Table[
        Print["starting m =", m , " at ", DateString[]];
        figureGdata[m ] = Table[ParallelTable[
              Join[{Ω/ω}, Re[E^(I ωt #) / E^(I getPhaseGamma[#]) &@
                    singleSPAdipoleDataNoIntrinsicPhase[m , Ω][[{1, 2}]]]]
              , {ωt, 0, 360 °, 2.5 °}], {Ω, Select[Keys[singleSPAdipoleData[1]][[
                  1 ;; ;; 1]], 12 ω ≤ # ≤ 61.5 ω &]}]
        , {m , {-1, 1, 0, 2}}]
    ]]

]
DateString[]
```

```
Fri 28 Apr 2017 21:51:59
starting m =-1 at Fri 28 Apr 2017 21:51:59
starting m =1 at Fri 28 Apr 2017 21:55:26
starting m =0 at Fri 28 Apr 2017 21:58:57
starting m =2 at Fri 28 Apr 2017 22:02:24
{832.137106, {4, 1189, 145, 3}}
Fri 28 Apr 2017 22:05:52
```

```
NotebookSave[]

Save[NotebookDirectory[] <> "data - 3D data for ellipsometry .txt", figureGdata]



Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction},
  {Ip, F, ω} = parameters ; κ = √(2 Ip) ; γ = κω/F;
  classifierFunction = Function[{t, τ, Ω}, Which@@Flatten[{Table[
          {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π/3]/(2 π/3) == k - 1], "A" <> ToString[k]]
          , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];
  (*selection = Query[Transpose] /@ Query[Transpose] [Association[ParallelTable[
          m →
            KeySort[ClassifyQuantumOrbits [saddlePoints[m ], classifierFunction,
                  sortingFunction, DiscardedLabels → {"Discard"}]] [[All, All, 1]]
          , {m , -1, 1}]]];*)

  Print[DateString[]];

  SetSharedFunction[figureG];
  SetSharedFunction[figureGcore];
  Column @AbsoluteTiming [ParallelTable[(*505s≈8m30s  for the 2p figure*)
        AbsoluteTiming [figureG[m ] = Show[
              {


                figureGcore[m ] = ParametricPlot3D [
                    BSplineFunction[figureGdata[m ]] [u^2.5, v], {u, 0, 1}, {v, 0, 1}
                    , PlotPoints → 75
                    , MaxRecursion → 4
                    , Mesh → None
                    , ColorFunctionScaling → False
                    , ColorFunction → Function[{HO, Dx, Dy, u, v},
                        Directive[Blend[{Lighter[Yellow, 0.6], ColorData["Rainbow"][
                              1 - (HO - 12)/(60 - 12)]}, 0.45], Opacity[0.45]]
                      ]
                  ],
                Table[
                  ParametricPlot3D [
                    Evaluate[
                      Join[{Ω/ω}, Re[e^(i ω t) singleSPAdipoleDataNoIntrinsicPhase
```

```
                              m  , Ω]〚{1, 2}〛]]]
                        , {ωt, 0, 2π}
                        , PlotStyle → Directive[Thickness[0.007],
                            ColorData["Rainbow"][1 - (Ω/ω - 12)/(60 - 12)]]
                      ]
                      , {Ω, Select[Keys[singleSPAdipoleData[1]]〚13 ;; ;; 72〛,
                          12ω < # ≤ 61.5ω &]}]
                    ,
                    Graphics3D[{
                        Inset[MaTeX["\\Omega /\\omega ", FontSize → tfs],
                          Scaled[{0.5, 0, -0.25}]],
                        Inset[MaTeX["\\begin{aligned}D_x(\\Omega ) \\\\
                                \\text{(arb.u.)}\\end{aligned}",
                            FontSize → tfs], Scaled[{1.07, 0.65, -0.3}]],
                        Inset[MaTeX["\\begin{aligned}D_y(\\Omega ) \\\\
                                \\text{(arb.u.)}\\end{aligned}", FontSize →
                              tfs], Scaled[{1.15, 0.85, 1.35}], Scaled[{1, 0.5}]]
                      }],
                    Graphics3D[{
                        Inset[MaTeX["\\mathrm  {(" <> {"a", "c", "b", "d"}〚m  + 2〛 <> ")}",
                            FontSize → 8], Scaled[{0, 0.15, 1.15}]]
                      }]
                  }
                  , PlotRange → {{12, 62}, 0.00017{-1, 1}, 0.00017{-1, 1}}
                  , PlotRangePadding → None
                  , BoxRatios → {2.75, 1, 1}
                  , ImageSize  → 194
                  , ViewPoint → {4, -3.6, 2}
                  , ViewVertical → {0, 0, 1}
                  , Background → None
                  , Axes → True
                  , Ticks → {
                      Join[{#, If[Divisible[#, 6], MaTeX[#, FontSize → tfs]], {0.03, 0}} &/@
                          Range[12, 62, 3], {#, "", {0.015, 0}} &/@Range[12, 62, 1]],
                      Join[{0.0001#, MaTeX[PaddedForm [#, {2, 1}], FontSize → tfs],
                              {0.03, 0}} &/@Range[-1.5, 1.5, 0.5],
                        {0.0001#, "", {0.015, 0}} &/@Range[-2, 2, 0.1]],
                      Join[{0.0001#, MaTeX[PaddedForm [#, {2, 1}], FontSize → tfs],
                              {0.03/2.75, 0}} &/@Range[-1.5, 1.5, 0.5],
                        {0.0001#, "", {0.015/2.75, 0}} &/@Range[-2, 2, 0.1]]
                    }
                ];(*Print["Finished m =",m ," at ",DateString[]];*)]
        , {m , {-1, 1, 0, 2}}]]
]
DateString[]
(*Magnify[Grid[{{figureG[-1],figureG[1]},{figureG[0],figureG[2]}}],2.5]*)
```

Thu 31 Aug 2017 18:30:22

```
1306.273014
{{1038.786312, Null}, {766.067284, Null}, {1305.582284, Null}, {728.761919, Null}}

Thu 31 Aug 2017 18:52:08

DateString[]
NotebookSave[]

Thu 31 Aug 2017 18:52:08

Save[NotebookDirectory[] <> "data - figure G core uncompressed .txt", figureGcore]

DateString[]
NotebookSave[]

Thu 31 Aug 2017 18:52:13

fGres = 700;
AbsoluteTiming[{
    FileByteCount[Export[
        FileNameJoin[{$OutputDirectory, "figureGa-polarization-ellipse-3d-p-.pdf"}],
        figureG[-1], ImageResolution → fGres]],
    FileByteCount[Export[FileNameJoin[{$OutputDirectory,
            "figureGb-polarization-ellipse-3d-p+.pdf"}],
        figureG[1], ImageResolution → fGres]],
    FileByteCount[Export[FileNameJoin[{$OutputDirectory,
            "figureGc-polarization-ellipse-3d-s.pdf"}],
        figureG[0], ImageResolution → fGres]],
    FileByteCount[Export[FileNameJoin[{$OutputDirectory,
            "figureGd-polarization-ellipse-3d-2p.pdf"}],
        figureG[2], ImageResolution → fGres]]}]
DateString[]
pdflatex[]

{19.308590, {197020, 224210, 111359, 212072}}

Thu 31 Aug 2017 18:52:33

pdflatex successful

DateString[]
NotebookSave[]

Thu 31 Aug 2017 18:52:35


NotebookSave[]

Save[NotebookDirectory[] <> "data - figure G uncompressed .txt", figureG]
DumpSave[NotebookDirectory[] <> "data - figure G compressed .mx ", figureG]

{figureG}

NotebookSave[]
```

Trajectories

SlideView overview over the harmonic plateau

```
Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction, data},
  {Ip, F, ω} = parameters ; κ = Sqrt[2 Ip] ; γ = κ ω / F;
  classifierFunction = Function[{t, τ, Ω}, Which @@ Flatten[{Table[
            {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t - τ], 2 π / 3] / (2 π / 3) == k - 1], "A" <> ToString[k]
            , {k, 1, 6}], {True, "Discard"}}]];
  sortingFunction =
    Function[list, SortBy[list, Function[Re[ω #[[1]] - Floor[ω Re[#[[1]] - #[[2]]], 2 π]]]]];
  (*selection=Query[Transpose]/@Query[Transpose][Association[ParallelTable[
            m →
              KeySort[ClassifyQuantumOrbits[saddlePoints[m ],classifierFunction,
                    sortingFunction,DiscardedLabels→{"Discard"}]][[All,All,1]]
            ,{m ,-1,1}]]];*)

  Block[{index = First[Keys[selection]],
      (*Ω=First[Nearest[Keys[selection[[1]]],18ω]],*)m  = 0},

    SlideView[Table[
        Column [{
            Row[{
                ParametricPlot [
                    Evaluate[Flatten[Table[
                            Tooltip[
                              part[
                                rotatingFrameTrajectory [
                                    Interpolation[
                                        Function[{t, τ}, {
                                        {{0, t - τ}, {1, t}},
                                        {{0, t - τ}, {1 / 3, Re[t - τ]},
                                        {2 / 3, Re[t]}, {1, t}},
                                        {{0, t - τ}, {1 / 2, Re[t - τ] + ⅈ Im  [t]}, {1, t}}
                                        }[[j]]] @@ selection[index, Ω, m  ]
                                        , InterpolationOrder → 1] [s]
                                    , Function[{t, τ},
                                        {t, t - τ}
                                      ] @@ selection[index, Ω, m  ]
                                  ][[{1, 2}]]
                              ]
```

```
            , part]
          , {j, 1, 3}, {part, {Im , Re}}], 1]]
    , {s, 0, 1}
    , PlotPoints→60
    , PlotStyle→{Lighter[Blue, 0.7], Lighter[Red, 0.7],
        Lighter[Blue, 0.35], Lighter[Red, 0.35], Blue, Red}
    , Frame  →True
    , Method→{"AxesInFront"→False}
    , ImageSize →700
    , PlotLabel→"Ω="<>ToString[Ω/ω]<>"ω"
    , PlotRange→{{-25, 10}, {-10, 10}}
  ] /. {Line[pts_]:→{Arrowheads[0.025], Line[pts], Arrow[pts]}}
,
Show[{
    ParametricPlot [
        Evaluate[Flatten[Table[
              Tooltip[
                part[
                  rotatingFrameVelocity [
                      Interpolation[
                      Function[{t, τ}, {
                      {{0, t-τ}, {1, t}},
                      {{0, t-τ}, {1/3, Re[t-τ]},
                      {2/3, Re[t]}, {1, t}},
                      {{0, t-τ}, {1/2, Re[t-τ]+ⅈIm [t]}, {1, t}}
                      }[[j]]] @@selection[index, Ω, m ]
                      , InterpolationOrder→1] [s]
                      , Function[{t, τ},
                      {t, t-τ}
                      ] @@selection[index, Ω, m ]
                  ] [[{1, 2}]]
                ]
                , part]
              , {j, 1, 3}, {part, {Im , Re}}], 1]]
        , {s, 0, 1}
        , PlotPoints→60
        , PlotStyle→{Lighter[Blue, 0.7], Lighter[Red, 0.7],
            Lighter[Blue, 0.35], Lighter[Red, 0.35], Blue, Red}
      ] /. {Line[pts_]:→{Arrowheads[0.025],
            Line[pts], Arrow[pts]}}
    ,
    Graphics[{PointSize[0.02],
        Function[{t, τ}, {
              Red, Point[Re[ps[t, t-τ][[{1, 2}]]]],
              Blue, Point[Im [ps[t, t-τ][[{1, 2}]]]]
            }] @@selection[index, Ω, m ]
      }]
  }
  , Frame  →True
  , Method→{"AxesInFront"→False}
  , ImageSize →{{700}, {20/35 700}}
  , PlotLabel→"Ω="<>ToString[Ω/ω]<>"ω"
```

```
                , PlotRange → {{-2.2, 2.2}, {-2.2, 2.2}}
              ]
            }]
          ,
          ParametricPlot [
              Evaluate[Table[
                  ReIm [ω×
                      Interpolation[
                          Function[{t, τ}, {
                                  {{0, t-τ}, {1, t}},
                                  {{0, t-τ}, {1/3, Re[t-τ]},
                                    {2/3, Re[t]}, {1, t}},
                                  {{0, t-τ}, {1/2, Re[t-τ] + ⅈ Im [t]}, {1, t}}
                                }[[j]] @@selection[index, Ω, m ]
                          , InterpolationOrder → 1] [s]
                  ]
                  , {j, 1, 3}]]
              , {s, 0, 1}
              , PlotPoints → 60
              , PlotStyle → {Lighter[Blue, 0.7], Lighter[Blue, 0.35], Blue}
              , Frame  → True
              , Method → {"AxesInFront" → False}
              , ImageSize  → 700
              , PlotLabel → "Ω=" <> ToString[Ω / ω] <> "ω"
              , PlotRange → {{1, 4}, {-1, 1}}
          ] /. {Line[pts_] :→ {Arrowheads [0.025], Line[pts], Arrow[pts]}}
        }]
      , {Ω, First[Nearest [Keys [selection[[1]]], #]] & /@Range [10 ω, 70 ω, 0.5 ω]}]]

    ]
  ]
```

Figure H

```
tfsH = 7;
sizeH= 72.5;

Block[{Ip, F, ω, κ, γ, saddles, classifierFunction, sortingFunction,
    steppingStones, timePath , lengthsList, totalPathLength, timePathMarkers },
  {Ip, F, ω} = parameters  ; κ = Sqrt[2 Ip] ; γ = κ ω / F;
  classifierFunction = Function[{t, τ, Ω}, Which@@Flatten[{Table[
          {And[0.95 < ω Re[τ] < 2.68, Floor[ω Re[t-τ], 2 π/3] / (2 π/3) == k-1], "A" <> ToString[k]
```

```
            , {k, 1, 6}], {True, "Discard"}}]];
sortingFunction=
  Function[list, SortBy[list, Function[Re[ω#[[1]] - Floor[ωRe[#[[1]] - #[[2]]], 2π]]]]];
selection= Query[Transpose] /@Query[Transpose] [Association[ParallelTable[
          m  → KeySort[ClassifyQuantumOrbits [saddlePoints[m ], classifierFunction,
                  sortingFunction, DiscardedLabels→ {"Discard"}]][[All, All, 1]]
          , {m  , -1, 1}]]];

Block[{index= First[Keys[selection]],
    m  = 0(*,Ω=First[Nearest[Keys[selection[[1]]],27ω]]*)},

  figureHinsetBackground= Show[{
        Graphics[{White, Opacity[0.85], Rectangle[{1.25, -0.68}, {4.0, 0.77}]}],
        Graphics[{GrayLevel[0.7], Line[{{1.25, 0}, {4.0, 0}}]}],
        Graphics[
          Table[{
              GrayLevel[0.6],
              AbsoluteThickness[0.85],
              Line[Table[
                  Function[{t, τ},
                      ReIm  [ω (time  /. {"tt" → t-τ, "t" → t, "τ" → τ})]
                    ]@@selection[index, Ω, m  ]
                  , {Ω, Keys[selection[index]][[1 ;; ;; 1]]}]]
            }, {time , {"t", "tt"}}]
        ],
        Graphics[Table[
            Inset[MaTeX["\\SI{"<>ToString[φ]<>"}{\\degree}", FontSize→ 9,
                Magnification→ 0.5], {φ°, -0.7}, Scaled[{0.5, 1}]]
            , {φ, 90, 210, 30}]],
        Graphics[Table[
            Inset[MaTeX[ToString[PaddedForm [φ, {2, 1}]], FontSize→ 9,
                Magnification→ 0.5], {1.24, φ}, Scaled[{1, 0.5}]]
            , {φ, -0.5, 0.5, 0.5}]]
      }
      , ImageSize  → 550
      , PlotRange → {{1.25, 4.0}, {-0.68, 0.77}}
      , Frame  → True
      , PlotRangePadding→ None
      , FrameTicks  → {
          {{#, ""} &/@Range[-1, 1, 0.5], {#, ""} &/@Range[-1, 1, 0.5]},
          {{#°, ""} &/@Range[0, 360, 30], {#°, ""} &/@Range[0, 360, 30]}
        }
      , ImagePadding  →
          {{Scaled[0.05], Scaled[0.0001]}, {Scaled[0.05], Scaled[0.0001]}}
    ];

  Table[
    steppingStones=
      Function[{t, τ}, {t-τ, Re[t-τ] + ⅈIm  [t], t}]@@selection[index, Ω, m  ];
    lengthsList= Abs[Rest[steppingStones] -Most[steppingStones]];
    totalPathLength = Total[lengthsList];
```

```
timePath = Interpolation[
    Transpose[{Accumulate [Join[{0}, lengthsList]], steppingStones}],
    InterpolationOrder→1];
timePathMarkers = Most[Rest[Range[0, totalPathLength, (*timeStep =*)2]]];

Row[{

    figureHinset[Round[Ω/ω]] = Show[{
          figureHinsetBackground,
          ParametricPlot [
              ReIm [ωtimePath [s]]
              , {s, 0, totalPathLength}
              , PlotPoints→60
              , PlotStyle→{Directive[Black, AbsoluteThickness[0.85]]}
              , PlotLabel→"Ω="<>ToString[Ω/ω]<>"ω"
            ] /. {Line[pts_] :→ {Arrowheads[{{0.09, 1}, {0.012, 0, Graphics[
                              {Disk[{0, 0}]}]}}], Line[pts], Arrow[pts]}},
          Graphics[{
              AbsolutePointSize[1.7],
              Point[ReIm [ω×timePath [#]] &/@timePathMarkers ]
            }]
        }
        , ImageSize →70
      ];

    figureHposition[Round[Ω/ω]] = Show[Join[{
            Graphics[{GrayLevel[0.7],
                Line[{{-15.3, 0}, {1.0, 0}}], Line[{{0, -2.85}, {0, 3.7}}]}]
            (*,PlotRange→{{-15.3,1},{-2.85,3.7}}*)
          }, Table[{
              ParametricPlot [
                  Tooltip[
                    part[
                      rotatingFrameTrajectory [timePath [s],
                          Apply[Function[{t, τ}, {t, t-τ}],
                            selection[index, Ω, m ]]][[{1, 2}]]
                    ]
                    , part]
                  , {s, 0, totalPathLength}
                  , PlotPoints→60
                  , PlotStyle→{Directive[part /.
                          {Re→Red, Im →Blue}, AbsoluteThickness[0.85]]}
                ] /. {Line[pts_] :→ {Arrowheads[0.035],
                      Line[pts], Arrow[pts]}},
              Graphics[{
                  AbsolutePointSize[2.2],
                  part /. {Re→Red, Im →Blue},
                  Point[
                    part[
                          rotatingFrameTrajectory [timePath [#],
                              Apply[Function[{t, τ}, {t, t-τ}],
                                selection[index, Ω, m ]]][[{1, 2}]]
```

```
                ] &/@timePathMarkers
            ]
          }]
      }, {part, {Im , Re}}], {
      Graphics[Inset[figureHinset[Round[Ω/ω]],
          Scaled[{0.06, 0.98}], Scaled[{0, 1}]]],
      Graphics[{
          Inset[MaTeX["x_{\\mathsf {R}}(t)", FontSize→7],
            Scaled[{0.47, -0.12}], Scaled[{0.5, 1}]],
          Inset[Rotate[MaTeX["y_{\\mathsf {R}}(t)", FontSize→7], 90°],
            Scaled[{-0.07, 0.5}], Scaled[{1, 0.5}]]
        }],
      Graphics[Inset[
          MaTeX["(\\mathrm  {"<>{"a", "c", "e", "g"}[[(Round[Ω/ω]-17)/10+1]]<>
              "})\\ \\ \\mathbf {r}_\\mathsf {R}(t), \\ \\Omega ="<>
              ToString[Round[Ω/ω]]<>"\\omega ", FontSize→9]
          , Scaled[{0.67, 0.97}], Scaled[{0.5, 1}]
        ]]
    }]
  , ImageSize → (Automatic → {Automatic , sizeH})
  , PlotRange→{{-15.3, 1}, {-2.85, 3.7}}
  , PlotRangePadding→None
  , PlotRangeClipping→False
  , Frame →True
  , Method→{"AxesInFront"→False}
  , AxesOrigin→{0, 0}
  , AxesStyle→GrayLevel[0.7]
  , FrameTicks →{{
        Join[{#, MaTeX[#, FontSize→tfsH], {0.016, 0}} &/@Range[-2, 4, 1],
          {#, "", {0.008, 0}} &/@Range[-4, 4, 0.5]],
        Join[{#, "", {0.016, 0}} &/@Range[-2, 4, 1],
          {#, "", {0.008, 0}} &/@Range[-4, 4, 0.5]]
      }, {
        Join[{#, MaTeX[#, FontSize→tfsH], {0.016, 0}} &/@Range[-15, 0, 5],
          {#, "", {0.008, 0}} &/@Range[-20, 5, 1]],
        Join[{#, "", {0.016, 0}} &/@Range[-15, 0, 5],
          {#, "", {0.008, 0}} &/@Range[-20, 5, 1]]
      }}
  , ImagePadding →{{Scaled[0.05], Scaled[0.001]},
      {If[Ω>40ω, Scaled[0.045], Scaled[0.0001]], Scaled[0.0010]}}
  ]
,

figureHvelocity[Round[Ω/ω]] = Show[
  Join[Table[{
        Graphics[{
            Arrowheads[{{0.035, 0.975}}],
            AbsoluteThickness[0.85],
            Blend[
```

```
            {part /. {Re → Red, Im  → Blue}, GrayLevel[0.9]}, 0.6],
          Arrow[{{0, 0},
              part[
                rotatingFrameVelocity [timePath [0],
                    Apply[Function[{t, τ}, {t, t - τ}],
                      selection[index, Ω, m ]]][[{1, 2}]]
              ]
            }](*
          ,Blend[{part/.{Re→Red,Im →Blue},GrayLevel[0.4]},0.7],
          Arrow[{{0,0},part[rotatingFrameVelocity [timePath [
                      totalPathLength],Apply[Function[{t,τ},{t,
                        t-τ}],selection[index,Ω,m ]]][[{1,2}]]]}]*)
        }]
    }, {part, {Im , Re}}],

  Table[{
      ParametricPlot [
          Tooltip[
            part[
              rotatingFrameVelocity [timePath [s],
                  Apply[Function[{t, τ}, {t, t - τ}],
                    selection[index, Ω, m ]]][[{1, 2}]]
            ]
            , part]
          , {s, 0, totalPathLength}
          , PlotPoints→ 60
          , PlotStyle → {Directive[part /.
                  {Re → Red, Im  → Blue}, AbsoluteThickness[0.85]]}
        ] /. {Line[pts_] :→ {Arrowheads[0.05],
              Line[pts], Arrow[pts]}},
      Graphics[{
          AbsolutePointSize[2.2],
          part /. {Re → Red, Im  → Blue},
          Point[
            part[
                  rotatingFrameVelocity [timePath [#],
                      Apply[Function[{t, τ}, {t, t - τ}],
                        selection[index, Ω, m ]]][[{1, 2}]]
                ] & /@timePathMarkers
          ]
        }]
    }, {part, {Im , Re}}], {
    Graphics[{
        Inset[MaTeX["v_{\\mathsf {R},x}(t)", FontSize→ 7],
          Scaled[{0.45, -0.13}], Scaled[{0.5, 1}]],
        Inset[Rotate[MaTeX["v_{\\mathsf {R},y}(t)", FontSize→ 7],
            90 °], Scaled[{1.108, 0.5}], Scaled[{0, 0.5}]]
      }],
    Graphics[Inset[

        MaTeX["(\\mathrm  {"<>{"b", "d", "f", "h"}[[(Round[Ω/ω] - 17)/10 + 1]]<>
```

```
                    "})\\ \\ \\mathbf {v}_\\mathsf {R}(t), \\ \\Omega ="<>
                    ToString[Round[Ω/ω]]<>"\\omega ", FontSize→9]
                , Scaled[{0.40, 0.97}], Scaled[{0.5, 1}]
              ]],
            Graphics[{
                If[Ω<20ω,
                  Inset[Grid[Table[{
                        Graphics[{part /. {Re→Red, Im →Blue}, AbsoluteThick
                              0.85], Line[{{0, 0}, {1, 0}}]},
                          PlotRange→{{0, 1}, 0.1{-1, 1}},
                          PlotRangePadding→None, ImageSize →10],
                        MaTeX["\\mathrm  {"<>ToString[part]<>"}",
                          FontSize→5.5]
                      }, {part, {Re, Im }}], Background→White,
                    Spacings→{0.5, 0}, Frame →True,
                    FrameStyle →Directive[Gray, Thickness[0.00001]]]
                  , Scaled[{0.97, 0.97}], Scaled[{1, 1}]]
                , ##&[]]
              }]
          }]
        , ImageSize →(Automatic →{Automatic , sizeH})
        , PlotRange→{{-1.5, 1.95}, {-0.55, 1.07}}
        , PlotRangePadding→None
        , PlotRangeClipping→False
        , Frame →True
        , Method→{"AxesInFront"→False}
        , Axes→True
        , AxesOrigin→{0, 0}
        , AxesStyle→GrayLevel[0.7]
        , FrameTicks →{{
              Join[{#, "", {0.02, 0}} &/@Range[-1, 1, 0.5],
                {#, "", {0.01, 0}} &/@Range[-2, 2, 0.2]],
              Join[{#, MaTeX[PaddedForm [#, {2, 1}], FontSize→tfsH],
                    {0.02, 0}} &/@Range[-1, 1, 0.5],
                {#, "", {0.01, 0}} &/@Range[-2, 2, 0.2]]
            }, {
              Join[{#, MaTeX[#, FontSize→tfsH], {0.02, 0}} &/@Range[-1, 2],
                {#, "", {0.01, 0}} &/@Range[-2, 3, 0.2]],
              Join[{#, "", {0.02, 0}} &/@Range[-1, 2],
                {#, "", {0.01, 0}} &/@Range[-2, 3, 0.2]]
            }}
        , ImagePadding →{{Scaled[0.001], Scaled[0.06]},
            {If[Ω>40ω, Scaled[0.045], Scaled[0.0012]], Scaled[0.0001]}}
      ]
    }]

  , {Ω, {17ω, 27ω, 37ω, 47ω}}]
]

]
```

```
Table[{
    FileByteCount[Export[FileNameJoin[{$OutputDirectory, "figureH" <>
            {"a", "c", "e", "g"}[[(HO - 17)/10 + 1]] <> "-position-" <> ToString[HO] <> ".pdf"}],
        figureHposition[HO], Background -> None]],
    FileByteCount[Export[FileNameJoin[{$OutputDirectory, "figureH" <>
            {"b", "d", "f", "h"}[[(HO - 17)/10 + 1]] <> "-velocity-" <> ToString[HO] <> ".pdf"}],
        figureHvelocity[HO], Background -> None]]
  }, {HO, {17, 27, 37, 47}}]
pdflatex[]
{{34290, 15126}, {35926, 14844}, {36236, 15552}, {38541, 17238}}

pdflatex successful
```